KLASIFIKASI GLIKOSILASI PADA SEQUENCE PROTEIN LISIN MENGGUNAKAN METODE *RANDOM FOREST*

Skripsi

Oleh

SILFIA FITRIYANA

1617051100

JURUSAN ILMU KOMPUTER

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS LAMPUNG

2023

ABSTRAK

KLASIFIKASI GLIKOSILASI PADA SEQUENCE PROTEIN LISIN MENGGUNAKAN METODE *RANDOM FOREST*

Oleh

Silfia Fitriyana

Glikosilasi pada sequence protein lisin proses pengklompokkan protein lisin yang terglikosilasi ke dalam kelompok-kelompok berdasarkan jenis glikosilasi yang terdapat pasa protein lisin. Klasifikasi glikosilasi pada sequence protein lisin dengan menggunakan metode random forest, tujuan dari penelitian ini untuk menganalisa sensitivitas, spesifisitas, akurasi, dan *matthew correlation coefficient* (MCC). Dataset protein lisin diperoleh dari data benchmark memiliki jumlah protein 620 dengan 214 data positif dan 406 data negatif serta memiliki panjang protein 15 sequence. Dataset dibagi menjadi 2, yaitu 90% data latih dan 10% data uji. Ekstraksi fitur yang digunakan protein descriptor menggunakan R package BioSeqClass versi 1.44.0, yaitu AA *Index*, CTD, dan PseAAC, dengan jumlah total 60 fitur. Hasil yang didapatkan diolah kembali menggunakan klasifikasi algoritme random forest dengan mtry 4, 8, dan 16 lalu ntree dipilih secara acak 250, 500, 750, dan 1000. Hasil yang didapatkan paling tinggi didapatkan pada pembagian dataset 90% data latih 10% data uji dengan *mtry* 42 ntree 1000 sebesar 89.97% sensitivitas, 92.79% spesifisitas, 80.76% MCC, dan 90.42% akurasi. Hasil yang didapatkan menggunakan ekstraksi fitur dan algoritme *random forest* bisa mengklasifikasikan protein lisin.

Kata Kunci : Glikosilasi, protein lisin, fitur ekstarksi, klasifikasi, random forest

ABSTRACT

GLYCOSYLATION CLASSIFICATION OF LYSINE PROTEIN SEQUENCE USING THE RANDOM FOREST METHOD

By

Silfia Fitriyana

Glycosylation in the lysine protein sequence is the process of grouping glycosylated lysine proteins into groups based on the type of glycosylation present in the lysine protein sequence. Classification of glycosylation in lysine protein sequences using the random forest method. The purpose of this study was to analyze the sensitivity, specificity, accuracy, and Matthew correlation coefficient (MCC). The lysine protein dataset obtained from benchmark data has a total protein of 620 with 214 positive data and 406 negative data and has a protein length of 15 sequences. The dataset is divided into 2, namely 90% training data and 10% test data. Feature extraction used protein descriptor using R package BioSeqClass version 1.44.0, namely AA Index, CTD, and PseAAC, with a total of 60 features. The results obtained were reprocessed using the random forest classification algorithm with mtry 4, 8, and 16, and then ntree was randomly selected 250, 500, 750, and 1000. The highest results obtained were obtained in the distribution of the dataset 90% training data 10% test data with mtry 42 ntree 1000 of 89.97% sensitivity, 92.79% specificity, 80.76% MCC, and 90.42% accuracy. The results obtained using feature extraction and random forest algorithms can classify lysine proteins.

Keywords: *Glycosylation, lysine protein, extraction features, classification, random forest*

KLASIFIKASI GLIKOSILASI PADA SEQUENCE PROTEIN LISINDENGAN MENGGUNAKAN METODE *RANDOM FOREST*

Oleh

SILFIA FITRIYANA

Skripsi

Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA ILMU KOMPUTER

Pada

Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2023

Judul Skripsi : **KLASIFIKASI GLIKOSILASI PADA SEQUENCE PROTEIN LISINDENGAN MENGGUNAKAN METODE *RANDOM FOREST***

Nama Mahasiswa : **Silfia Fitriyana**

Nomor Pokok Mahasiswa : **1617051100**

Program Studi : **Ilmu Komputer**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**

MENYETUJUI

1. Komisi Pembimbing

Favorisen R. Lumbanraja, Ph.D.	**Yohana Tri Utami,S.Kom., M.Kom.**
NIP. 19830110 200812 1 002	NIP. 19900110 201903 2 010

2. Ketua Jurusan Ilmu Komputer

Didik Kurniawan, S.Si., M.T.
NIP. 19800419 200501 1 004

MENGESAHKAN

1. Tim Penguji

Ketua : **Favorisen R. Lumbanraja, Ph.D.** .....

Sekretaris : **Yohana Tri Utami, S.Kom., M.Kom.** .....

Penguji
Bukan Pembimbing : **Prof. Admi Syarif, Ph.D.** .....

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

Dr. Eng. Suripto Dwi Yuwono, S.Si., M.T.
NIP. 197407052000031001

Tanggal Lulus Ujian Skripsi : **10 Januari 2023**

PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul "**Klasifikasi Glikosilasi Pada Sequence Protein Lisin Menggunakan Metode *Random Forest***" ini merupakan hasil karya sendiri dan bukan hasil karya orang lain. Semua tulisan yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi ini merupakan hasil salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi berupa pencabutan gelar akademik yang telah saya terima.

Bandar Lampung, 10 Januari 2023

Silfia Fitriyana
1617051100

RIWAYAT HIDUP

Penulis dilahirkan pada 22 Januari 1999 di Kota Bandar Lampung, merupakan putri pertama dari pasangan Bapak Usman dan Ibu Dandayati. Penulis menyelesaikan pendidikan formal pertamanya di Taman Kanak-Kanak (TK) Amalia Tanjung Senang Bandar Lampung pada tahun 2004, melanjutkan sekolah dasar di SD Negeri 2 Tanjung Seneng Bandar Lampung dan selesai pada tahun 2010. Kemudian, penulis melanjutkan pendidikan sekolah menegah pertama di SMP Negeri 19 Bandar Lampung yang diselesaikan pada tahun 2013, serta menyelesaikan pendidikan sekolah menengah kejuruan pada tahun 2016 di SMK Negeri 2 Bandar Lampung dengan Jurusan Teknik Komputer Jaringan.

Pada tahun 2016, penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung dengan jalur Seleksi Bersama Masuk Perguruan Tinggi Negeri (SBMPTN). Selama menjadi mahasiswa, penulis mengikuti beberapa kegiatan antara lain:

1. Anggota Abacus Himpunan Mahasiswa Jurusan Ilmu Komputer periode 2016/2017.
2. Anggota bidang Eksternal Jurusan Ilmu Komputer (HIMAKOM) periode 2017.
3. Pada bulan Desember 2018 penulis melaksanakan kerja praktik di.Dinas Komunikasi, Informatika, Dan Statistik Provinsi Lampung
4. Pada bulan Juli 2019 penulis melaksanakan KKN di Desa Mekar Karya, KecamatanWaway Karya, Kabupaten Lampung Utara.
5. Pada bulan Oktober 2020 penulis mengikuti ujian sertifikasi yang diselenggarakan oleh BNSP dengan kualifikasi atau kompetensi sebagai *Junior Office Operator*.

MOTTO DAN PERSEMBAHAN

“Di tengah kesulitan tedapat kesempatan”
-Albert Einstein-

Skripsi ini saya persembahkan kepada:

Allah SWT yang telah memberikan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi ini

Terima kasih kepada kedua orang tua, Ayah dan Ibu, yang selalu memberikan kasih sayang, dukungan, semangat, dan doa.

Terima kasih kepada Bapak dan Ibu dosen di Jurusan Ilmu Komputer yang senantiasa membagikan ilmu dan memberikan nasihat yang memotivasi. Terima kasih kepada teman-teman Ilmu Komputer 2016 yang juga selalu mendukung dan berjuang bersama dalam meraih cita-cita

SANWANCANA

Puji syukur kehadirat Allah Subhanahu wa ta'ala yang telah melimpahkan rahmat, hidayah serta inayah-Nya sehingga penulis dapat menyelesaikan skripsi ini berjudul "Klasifikasi Glikosilasi Pada Sequence Protein Lisin Menggunakan Metode *Random Forest*". Tidak lupa shalawat serta salam kepada Nabi Muhammad Shallahu 'alaihi wasallam, yang kita nanti-nantikan syafaatnya di yaumil akhir kelak.

Penulis mengucapkan terimakasih kepada semua pihak yang telah membantu dan memiliki peran besar dalam penyusunan skripsi ini, yaitu:

1. Kedua orang tuaku tercinta, ayahku Usman dan ibuku Dandayati yang selalu memberikan semua dukungan, kasih sayang dan segala doa yang tiada hentinya.
2. Adekku M Fadli Ramadhan, yang selalu mendukung dan membantu kakakmu ini dalam menyelesaikan skripsi, dan selalu menemani kakak dalampenyelesaian skripsi ini.
3. Bapak Favorisen R. Lumbanraja, Ph.D., selaku dosen pembimbing utama atas kesediaannya dan kesabarannya untuk memberikan dukungan, bimbingan,kritik, dan saran dalam proses penyelesaian skripsi.
4. Ibu Yohana Tri Utami, S.Kom., M.Kom., selaku dosen pembimbing kedua atas kesediaannya dalam memberikan bimbingan, nasihat, juga kritik, dan saran selama proses pengerjaan skripsi.
5. Bapak Prof. Admi Syarif, Ph.D., selaku dosen pembahas skripsi yang telah memberikan saran dan masukan guna penyempurnaan penulisan skripsi.
6. Bapak Dekan Dr. Eng. Suripto Dwi Yuwono, S.Si., M.T., selaku Dekan FMIPA Universitas Lampung.
7. Bapak Didik Kurniawan, S.Si., MT., selaku Ketua Jurusan Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas

Lampung.

8. Bapak Dr. rer. nat. Akmal Junaidi, M.Sc., selaku Sekertaris Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung.
9. Bapak dan Ibu Dosen Jurusan Ilmu Komputer Universitas Lampung yang telah memberikan ilmu dan pengetahuan hidup selama penulis menjadi mahasiswa.
10. Refina Dhea, Novi Anisa, Maya Asterita, Dinda Fitrialoka, Monita Dwi, Elva Juasta dan Friska Daesy yang telah memberkan motivasi serta semangatdalam menyusun skripsi ini.
11. Semua pihak yang secara langsung maupun tidak langsung yang telah membantu menyelesaikan skripsi ini.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna, akan tetapi semoga skripsi ini dapat membawa manfaat dan keberkahan bagi perkembangan ilmu pengetahuan terutama bagi semua civitas Ilmu Komputer Universitas Lampung.

Bandar Lampung, 10 Januari 2023

Silfia Fitriyana
1617051100

DAFTAR ISI

Halaman

DAFTAR ISI.....**ii**
DAFTAR TABEL.....**vii**
DAFTAR GAMBAR.....**viii**
DAFTAR KODE PROGRAM.....**ix**
I. PENDAHULUAN.....**1**
1.1 Latar Belakang.....1
1.2 Rumusan Masalah.....3
1.3 Batasan Masalah.....3
1.4 Tujuan Penelitian.....3
1.5 Manfaat Penelitian.....3
II. TINJAUAN PUSTAKA.....**4**
2.1. Penelitian Terdahulu.....4
2.2 Protein.....7
2.2.1 Struktur Primer.....7
2.2.2 Struktur Sekunder.....8
2.2.3 Struktur Tersier.....8
2.2.4 Struktur Kuarterner.....8
2.3 Asam Amino.....8
2.4 Post Translational Modifications (PTM).....9
2.5 Glikosilasi.....9
2.6 Fitur Ekstraksi.....10
2.6.1 AA *index*.....11
2.6.2 *Composition, Transition, dan Distribution* (CTD).....11
2.6.3 *Pseudo Amino Acid Composition* (PseAAC).....12
2.7. *Machine Learning*.....13
2.7.1. *Machine Learning* Berdasarkan tipe data.....13

2.8. *Random Forest*.....13
2.9. Tingkat Misklasifikasi (*Out of Bag Error*) 19
2.10. *Variable Importance*.....19
2.10.1. *Mean Decrease Accuracy* (MDA).....20
2.10.2. *Gini Impruty Index* (GII).....20
2.11. *Cross Validation*.....20
2.11.1. *K-Fold Cross-Validation* 21
2.11.2. *Leave-One-Out Cross-Validation (LOOCV)*.....22
2.11.3. *Hold-Out Validation* 23
2.12. *Confusion matrix* 24
III. METODOLOGI PENELITIAN.....27
3.1 Tempat dan Waktu Penelitian 27
3.1.1 Tempat Penelitian.....27
3.1.2 Waktu Penelitian 27
3.2. Data dan Alat.....28
3.2.1. Data.....28
3.2.2 Alat.....28
3.3 Alur Kerja Penelitian.....28
3.3.1 Data 29
3.3.2 *Preprocessing*.....30
3.3.3 *Feature Extraction*/ Ekstraksi fitur.....30
3.3.4 *Classification*.....30
3.3.5 Pengujian 30
3.3.6 Evaluasi 30
IV. PEMBAHASAN.....27
4.1. Tahapan Percobaan.....31
4.1.1. Pembersihan dan Pembagian Data 31
4.1.2. *Cleaning* Data.....31
4.1.3. *Import* data dari file TXT 32
4.2. *Feature Extraction*.....32
4.2.1. *Composition/Transition/Distribution* (CTD).....32
4.2.2. AA *index*.....33

4.2.3. *Pseudo Amino Acid Composite* 34
4.3 Menggabungkan *Feature Extraction* dan Memberi Label 34
4.4 *Variable Importance*..... 35
4.5 Pembagian Data Menggunakan *k-Fold Cross Validation*.....36
4.6 Pemilihan *Mtry* dan *Ntree*.....36
4.7 Klasifikasi dengan *Random Forest*.....37
4.8 Pengujian Hasil Klasifikasi 37
4.8.1 Pengaruh Fitur Berdasarkan *Variable Importance*.....37
4.8.2 Pengujian 90% *training* 10% *testing*.....40
a. Pengujian *Mtry* 4.....40
b. Pengujian *Mtry* 8 48
c. Pengujian *Mtry* 16.....56
4.9 Pembahasan.....64
4.9.1 Hasil Tingkat Misklasifikasi (*out of bag error*).....65
4.9.2 Hasil Pengujian Klasifikasi.....66
4.9.3 Perbandingan Dengan Penelitian Sebelumnya.....68
V. PENUTUP 69
5.1 Simpulan.....69
5.1 Saran.....69
DAFTAR PUSTAKA 69

DAFTAR TABEL

Tabel Halaman

1. Penelitian Terdahulu Terkait Klasifikasi Glikosilasi 4
2. Macam-macam Asam Amino (Akram et al., 2011) 9
3. Contoh Soal *Random Forest* 17
4. Perhitungan Soal *Random Forest* 17
5. Nilai Gini Split Pada Setiap Kemungkinan 18
6. *Confusion Matrix* (Kuhn & Johnson, 2013) 25
7. Data *Sequence* Protein Glikosilasi 28
8. Total Keseluruhan Variabel 35
9. 10-*fold out of bag error mtry* 4 dan *ntree* 250 41
10. 10-*fold confusion matrix mtry* 4 dan *ntree* 250 41
11. Hasil pengujian klasifikasi 10-*fold mtry* 4 dan *ntree* 250 42
12. 10-*fold out of bag error mtry* 4 dan *ntree* 500 43
13. 10-*fold confusion matrix mtry* 4 dan *ntree* 500 43
14. Hasil pengujian klasifikasi 10-*fold mtry* 4 dan *ntree* 500 44
15. 10-*fold out of bag error mtry* 4 dan *ntree* 750 45
16. 10-*fold confusion matrix mtry* 4 dan *ntree* 750 45
17. Hasil pengujian klasifikasi 10-*fold mtry* 4 dan *ntree* 750 46
18. 10-*fold out of bag error mtry* 4 dan *ntree* 1000 47
19. 10-*fold confusion matrix mtry* 4 dan *ntree* 1000 47
20. Hasil pengujian klasifikasi 10-*fold mtry* 4 dan *ntree* 1000 48
21. 10-*fold out of bag error mtry* 8 dan *ntree* 250 49
22. 10-*fold confusion matrix mtry* 8 dan *ntree* 250 49
23. Hasil pengujian klasifikasi 10-*fold mtry* 8 dan *ntree* 250 50
24. 10-*fold out of bag error mtry* 8 dan *ntree* 500 51
25. 10-*fold out of bag error mtry* 8 dan *ntree* 500 (lanjutan) 52
26. 10-*fold confusion matrix mtry* 8 dan *ntree* 500 52

27. Hasil pengujian klasifikasi 10-*fold mtry* 8 dan *ntree* 500 52
28. 10-*fold out of bag error mtry* 8 dan *ntree* 750.....53
29. 10-*fold confusion matrix mtry* 8 dan *ntree* 750 53
30. Hasil pengujian klasifikasi 10-*fold mtry* 8 dan *ntree* 750.....54
31. 10-*fold out of bag error mtry* 8 dan *ntree* 1000.....54
32. 10-*fold out of bag error mtry* 8 dan *ntree* 1000 (lanjutan) 55
33. 10-*fold confusion matrix mtry* 8 dan *ntree* 1000 55
34. Hasil pengujian klasifikasi 10-*fold mtry* 8 dan *ntree* 1000.....56
35. 10-*fold out of bag error mtry* 16 dan *ntree* 250.....57
36. 10-*fold confusion matrix mtry* 16 dan *ntree* 250 57
37. Hasil pengujian klasifikasi 10-*fold mtry* 16 dan *ntree* 250.....58
38. 10-*fold out of bag error mtry* 16 dan *ntree* 500.....59
39. 10-*fold confusion matrix mtry* 16 dan *ntree* 500 59
40. Hasil pengujian klasifikasi 10-*fold mtry* 16 dan *ntree* 500.....60
41. 10-*fold out of bag error mtry* 16 dan *ntree* 750.....61
42. 10-*fold confusion matrix mtry* 16 dan *ntree* 750 61
43. Hasil pengujian klasifikasi 10-*fold mtry* 16 dan *ntree* 750.....62
44. 10-*fold out of bag error mtry* 16 dan *ntree* 1000.....63
45. 10-*fold confusion matrix mtry* 16 dan *ntree* 1000 63
46. Hasil pengujian klasifikasi 10-*fold mtry* 16 dan *ntree* 1000.....64
47. Hasil keseluruhan *out of bag error*.....65
48. Hasil keseluruhan pengujian klasifikasi 66
49. Hasil keseluruhan pengujian klasifikasi (lanjutan).....67
50. Perbandingan Kinerja Klasifikasi 68

DAFTAR GAMBAR

Gambar Halaman

1. Tingkatan Struktur Protein (Azhar, 2016) 8
2. Jenis Utama Glikosilasi Pada Manusia (Reily et al., 2019) 10
3. Contoh soal *Random Forest* 16
4. Tree soal *Random Forest* variabel pertama 18
5. Tree soal *Random Forest* seluruh variabel 19
6. Ilustrasi *k-fold* untuk k=10 (Lukito dan Chrismanto, 2015) 22
7. Ilustrasi *Leave-One-Out Cross-Validation* 23
8. Tahapan proses klasifikasi dengan model *Random Forest* 29
9. *Mtry* 4 *variable importance* 38
10. *Mtry* 8 *variable importance* 39
11. *Mtry* 16 *variable importance* 40
12. Grafik tingkat misklasifikasi 10-*fold (out of bag error*). 66
13. Hasil pengujian klasifikasi menggunakan ntree 250, 500, 750, dan 1000... 57

DAFTAR KODE PROGRAM

Kode Program Halaman

1. Kode program *import* data ke Rstudio.....32
2. Kode program *feature extraction* CTD 33
3. Kode program *feature extraction* AA*Index*.....33
4. Kode program *feature extraction* PseAAC 34
5. Kode program menggabungkan *feature extraction*.....34
6. Kode program memberi label kelas.....35
7. Kode program *variable importance* 35
8. Kode program pembagian data *training* dan *testing*.....36
9. Model klasifikasi *random forest*.....37
10. Kode program prediksi dengan *random forest*.....37

I. PENDAHULUAN

1.1. Latar Belakang

Post Translational Modifications (PTM) terdiri dalam modifikasi kovalen dari asam amino dari urutan protein primer dan memiliki efek untuk menciptakan susunan yang lebih besar dari spesies protein. PTM mempunyai peran penting dalam mengatur banyak fungsi biologis sebagai lokalisasi protein dalam sel, stabilitas protein, dan regulasi aktivitas enzimatik. Hingga saat ini, lebih dari 90.000 PTM individu adalah yang terdeteksi menggunakan analisis biokimia dan biofisik yang sering terjadi dalam skala besar disebuah sistem biologis menghasilkan wawasan tentang peran dan relevansinya dengan keadaan penyakit dan memberikan proteomik keunggulan yang unik (Audagnotto & Dal Peraro, 2017). Berbagai jenis respons biologis tergantung keinginan dan keadaan *metabolisme*, protein akan memungkinkan untuk *difosforilasi*, *glikosilasi*, *asetilasi*, *ubiquitinated*, *farnesylated*, *sulphated* dan terkait dengan jangkar GPI (Pascovici et al., 2019).

Salah satu dari modifikasi pasca translasi yaitu glikosilasi merupakan reaksi yang terjadi antara protein dan glukosa pada konsentrasi tinggi, reaksi ini disebut juga reaksi *Maillard* (Apriani, Suhartono, Akbar, & Mangkurat, 2011). Reaksi glikosilasi atau reaksi *Maillard* adalah reaksi antara gugus amina protein dengan gugus aldehid dari glukosa yang dapat membentuk produk-produk reaktif, yang selanjutnya dapat memodifikasi protein. Reaksi ini dicirikan dengan terjadinya pencokelatan nonenzimatik antara gula pereduksi dan asam amino bebas yang reaktif dari protein (Suhartono & Setiawan, 2008).

Glikosilasi akan menimbulkan reaksi non enzimatik yang umumnya terjadi *modifikasi post translational* protein yang diinduksi secara spontan oleh

kondensasi turunan glukosa dan senyawa metabolik intermediet dengan kelompok *amin* bebas pada residu *lisin* atau *arginin*. Langkah pertama reaksi *Maillard* adalah pembentukan kompleks inklusi basa *schiff* menjadi protein. Produk awal glikasi ini mengalami penyusunan kembali yang reversibel menjadi bentuk produk komplek inklusi *Amadori*. Basa *schiff* dan produk *Amadori* kemudian mengalami penyusunan kembali, oksidasi, dan atau pengeringan melalui jalur kimia untuk memproduksi kompleks inklusi yang *ireversibel* menjadi protein, yaitu *Advanced Glycation End Products* (AGEs) (Apriani et al., 2011).

Berdasarkan faktor penyebab diatas, diperlukan suatu pengklasifikasi yang diharapkan dapat membantu dunia medis. Hal ini dapat dilakukan dengan metode klasifikasi pada data *mining* yang sesuai informasi dalam membantu mencapai hasil dari pengklasifikasi tersebut. Pada penelitian glikolasi terdapat beberapa contoh ialah, oleh Xu, Y., et al., (2016) tentang *Identifying protein lysine glycation through sequences* menghasilkan tingkat akurasi, sensitif, spesifik, AUC dan MCC adalah 68,91%, 58,74%, 73,99%, 0,7258 dan 0,3198.

Penelitian ini mengklasifikasi suatu data yang disusun secara sistematis ke dalam suatu kelompok dapat mengetahui suatu individu berada pada kelompok tertentu, pada penelitian ini metode *classifier* yang digunakan adalah *random forest*. *Random forest* adalah metode klasifikasi yang banyak digunakan yang menggabungkan ide *bagging* pohon klasifikasi dan pengacakan fitur subset, didasarkan pada keputusan pohon dan menggunakan ide-ide agregasi (Breiman 2001). Skema ini untuk membangun prediktor *ensemble* dengan seperangkat pohon keputusan yang tumbuh di *subspaces* data yang dipilih secara acak. Meskipun minat dan penggunaan praktis, telah ada sedikit eksplorasi sifat statistik *random forest*, dan sedikit yang diketahui tentang kekuatan matematika yang mendorong algoritma. Pengklasifikasi *random forest* telah terbukti menghasilkan pohon dengan bias rendah dan memiliki korelasi rendah antara masing-masing pohon, menciptakan pengklasifikasi efisien, terutama pada data berdimensi tinggi dan struktur pohon secara eksplisit menangkap interaksi di antara fitur.

Klasifikasi Glikosilasi Pada *Sequence* Protein *Lisin* Menggunakan *Random Forest* penelitian ini lebih berfokus pada urutan atau *sequence* protein *Lisin* dengan tujuan mengklasifikasikan *sequence* protein *Lisin* mana yang telah terglikosilasi dan tidak terglikosilasi dengan menggunakan metode *random forest*.

1.2. Rumusan Masalah

Rumusan masalah dari penilitian ini berdasarkan pemaparan latar belakang, maka rumusan masalah dari penelitian ini bagaimana menentukan akurasi glikosilasi pada *sequence* protein lisin dengan klasifikasi menggunakan metode *random forest*.

1.3. Batasan Masalah

Batasan masalah pada penelitian ini adalah sebagai berikut:

1. Data yang digunakan yaitu data protein yang didapat dari studi literature.
2. Metode klasifikasi yang digunakan adalah metode *random forest*.

1.4. Tujuan Penelitian

Tujuan dari penelitian ini berdasarkan rumusan masalah adalah untuk mengukur tingkat keakuratan kinerja algoritma *random forest* dalam melakukan pengklasifikasi menghasilkan berupa *accuracy, sensitivitas, spesifisitas* dan MCC.

1.5. Manfaat Penelitian

Manfaat pada penelitian ini adalah mengetahui hasil dan evaluasi dari kinerja klasifikasi glikosilasi pada *sequence* protein lisin menggunkan algoritma random forest.

II. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Dalam melakukan suatu penelitian diperlukan penelitian yang sudah ada sebelumnya, yang berkaitan dengan penelitian tersebut. Referensi atau tinjauan pustaka dalam penelitian ini ditujukan pada Tabel 1

Tabel 1. Penelitian Terdahulu Terkait Klasifikasi Glikosilasi.

No	Penelitian	Data	Metode	Hasil
1	*Identifying Protein Lysine Glycation Through Sequences* metode *Support Vector Machine*(SVM) (Yun Xu et al., 2016)	***Benchmark Datase*** Jumlah: 1165 Negatif: 942 Positif: 223	*Support Vector Machine*	Akurasi : 68,91% Sensitivitas : 58,74% Spesifisitas : 73,99% MCC : 0,38%
2	Implementasi Metode *Random Forest* Untuk Prediksi Metilasi Pada Sekuens Protein Arginin. (Lumbanraja et al.,2019)	**Data Uji** Positif:162 Negatif:153 **Data Independent** Positif: 87 Negatif: 1.007	*Random Forest*	Menunjukkan hasil yang baik dengan akurasi 98,08% dan 93,76%.
3	Prediksi Glikosilasi Pada N-, C- dan O- Dalam Proteom ManusiaMengguna kan Fitur Seleksi Mrmr Dan Algoritme *Support Vector Mechine* (Lumbanraja et al., 2021)	***Benchmark Datase*** Jumlah: 2701 Negatif: 1973 Positif: 908 **Independent Dataset** Jumlah: 667 Negatif: 425 Positif 225	*Support Vector Machine*	***Benchmark*** C- : 80% N-: 66,01% O-: 88,52% ***Independent*** C- : 82,50% N-: 60,89% O-: 79,31%

Penelitian terdahulu digunakan sebagai acuan dan perbandingan dalam penelitian ini. Beberapa penelitian yang digunakan adalah sebagai berikut:

2.1.1 Penelitian yang terdahulu dilakukan oleh Yun Xu et al., (2016) dalam penelitian ini klasifikasi glikosilasi pada *sequence* protein lisin**,** dalam proses klasifikasi untuk pengukuran *true positive* (TP) untuk menghitung jumlah dengan benar prediksi peptida dalam *dataset* positif, *true negative* (TN) untuk menghitung jumlah peptida yang tidak terglikasi diprediksi dengan benar, *false positive* (FP) untuk menghitung non-terglikasi salah diprediksi sebagai peptida terglikasi, dan *false negative* (FN) untuk menghitung peptide terglikasi dan memprediksi sebagai peptida non-terglikasi yang akan mendapatkan hasil Sen, Spe, Acc, dan MCC.

Dalam penelitian ini memakai *benchmark dataset*¸ sampel peptide yang digunakan 223 + 942 = 1165 dimana 223 adalah peptida terglikasi yang termasuk dalam bagian positif dan sampel negatif memiliki 942 yang dipilih secara acak dari bagian negatif. Model klasifikasi dapat dievaluasi dengan empat strategi validasi yang banyak digunakan, yaitu tes *dataset* independen, *k-fold cross validation test and leave-one-out* (LOO). Validasi silang menghasilkan tiga metode yaitu GlyNN, PreGly dan Gly-PseAAC. Contoh yang digunakan pada penelitian ini adalah protein P00698 dan protein P01742. Penelitian ini dapat memberikan informasi urutan dan kecenderungan posisi asam amino spesifik digunakan untuk membangun fitur. Dalam LOO cross-validasi akurasi, sensitivitas, spesifisitas, AUC dan MCC adalah masing-masing 68,91%, 58,74%,73,99%, 0,7258 dan 0,3198.

2.1.2 Penelitian terdahulu dilakukan oleh Lumbanraja et al., (2019) dalam penelitian ini melakukan klasifikasi metilasi pada *sequence* protein *arginine***,** dalam proses klasifikasi untuk pengukuran *true positive* (TP) untuk menghitung jumlah dengan benar prediksi peptida dalam *dataset dataset* positif, *true negative* (TN) untuk menghitung jumlah peptida yang tidak terglikasi diprediksi dengan benar, *false positive*

(FP) untuk menghitung non-terglikasi salah diprediksi sebagai peptida terglikasi, dan *false negative* (FN) untuk menghitung peptida terglikasi dan memprediksi sebagai peptida non-terglikasi yang akan mendapatkan hasil Sen, Spe, Acc, dan MCC. Data yang digunakan menggunakan data *testing* dan *independent*. Proses penelitian yang dilakukan pertama adalah *preprocessing* dengan menghapus data asam amino yang tidak terdeteksi. Tahap selanjutnya itu fitur ekstraksi, model prediksi dengan *random forest*, pengujian dengan *cross validation*, dan yang terakhir yaitu membandingkan hasildengan penelitian sebelumnya. Hasil yang didapatkan adalah metode *random forest* menunjukkan hasil yang baik dengan akurasi 98,08% dan 93,76%. Namun, pada data *testing* didapatkan hasil akurasi yang lebih rendah dari penelitian Kumar et al., yakni sebesar 80,32%.

2.1.3 Penelitian terdahulu dilakukan oleh Lumbanraja et al., (2021) dalam penelitian ini melakukan menmggunakan 2 jenis data yaitu *independent* dan *benchmark*. Hasil akurasi pengujian data menggunakan pemilihan fitur sebanyak 25 kolom menggunakan MRMR pada data *independent* N sebesar 67,5% pada kernel polynomial, sedangkan pada data *independent* C sebesar 82,5% terletak di kernel RBF dan pada data *independent* O sebesar 85,83% pada kernel RBF. Data selanjutnya, yaitu data *benchmark* N sebesar 65,83% terdapat pada kernel RBF, sedangkan pada data *benchmark* C sebesar 69% dikernel sigmoid dan pada data *benchnark* O akurasi terbesar terdapat pada kernel RBF sebesar 87,02%.

2.2. Protein

Protein adalah komponen makro molekul utama yang dibutuhkan makhluk hidup, fungsi protein yang lebih diutamakan untuk sintesis protein-protein baru sesuai kebutuhan tubuh (Mipa, Semarang, Mulya, Lemong, & Barat, 2017). Struktur protein terdapat 21 jenis asam amino yang ada di alam dapat dibentuk protein dengan jenis yang tidak terbatas (Virus et al., 2019). Protein

berfungsi sebagai katalisator, sebagai pengangkut dan penyimpan molekul lain seperti oksigen, mendukung secara mekanis sistem kekebalan (imunitas) tubuh, menghasilkan pergerakan tubuh, sebagai transmitor gerakan syaraf dan mengendalikan pertumbuhan dan perkembangan (Katili, 2009). Proses terbentuknya protein dinamakan dengan sintesis protein. Sintesis protein DNA menjadi sumber genetik yang sangat penting. Pada DNA terdapat huruf A, C, T, dan G yaitu nukleotida pada DNA. Di dalam gen, penyandi protein adalah kumpulan-kumpulan nukleotida yang tersusun menjadi sebuah bentuk kode tiga huruf yang disebut dengan kodon. Kumpulan kodon akan bertranskripsi menjadi mRNA yang akan membentuk kode genetik. Informasi genetik inimengalir dari DNA ke RNA dan ke protein. Proses sintesis protein ini melibatkanRNA karena pembentukan protein berada pada ribosom, sedangkan informasi genetik dimulai dari DNA yang berada di nukleus.

Sekuens nukleotida sebuah molekul mRNA mengandung serangkaian kodon yang menentukan sekuens asam amino protein yang dikodenya (Murray et al., 2018). Sifat modular sintesis protein dapat dilihat dalam konsep tingkat struktur protein.Terdapat 4 struktur protein yang dapat dilihat pada Gambar 1 menurut (Azhar , 2016).

2.2.1. Struktur Primer

Residu asam amino yang menentukan struktur primer protein. Residu asam amino selalu dituliskan dari gugus terminal amino ke gugus terminal karboksil.

2.2.2. Struktur Sekunder

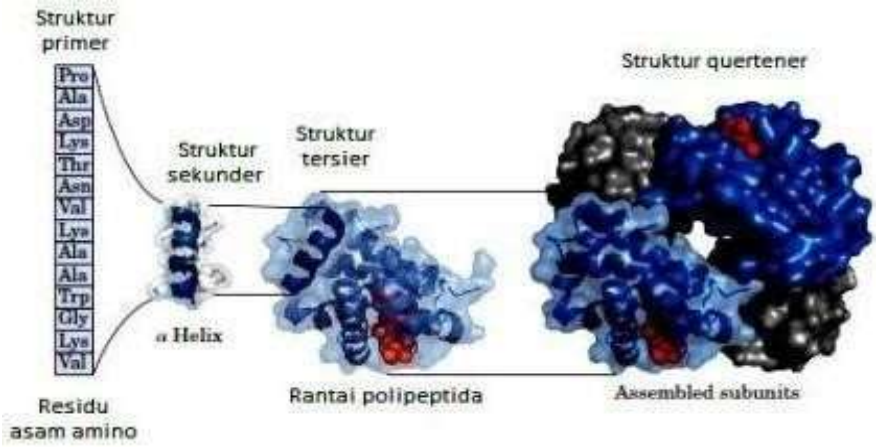
Struktur sekunder dibentuk karena adanya ikatan hidrogen amida dan oksigenkarbonil dari rangkaian peptida. Struktur sekunder utama meliputi αheliks dan β-strands (termasuk β-sheets).

2.2.3. Struktur Tersier

Perakitan unit struktural sekunder menjadi unit fungsional yang lebih besar seperti polipeptida matang dan domain komponennya (Murray et al., 2018).

2.2.4. Struktur Kuarterner

Jumlah dan jenis unit polipeptida protein oligomerik dan susunan spasialnya (Murray et al., 2018).



Gambar 1. Tingkatan Struktur Protein (Azhar, 2016).

2.3. Asam Amino

Protein penting bagi tubuh manusia, komponen utama dalam penyusun protein adalah asam amino yang berfungsi sebagai metabolisme dalam tubuh. Asam amino adalah unit dasar protein yang mengandung gugus amino dan gugus karboksilat, asam amino memainkan peran utama dalam mengatur berbagai proses terkait dengan ekspresi gen. Asam amino dibagi dua kelompok, yaitu asam amino esensial dan non-esensial. Asam amino esensial adalah asam amino yang tidak dapat dibuat oleh tubuh dan harus diperoleh dari makanan sumber protein, sedangkan asam amino non esensial merupakan asam amino yang dapat dibuat oleh tubuh manusia. Asam amino terdiri atas unsur-unsur karbon, hidrogen, oksigen, dan nitrogen. Unsur nitrogen adalah unsur utama protein sebanyak 16% dari berat protein. Asam amino digunakan dalam pembentukan protein, jika asam amino kurang, maka sintesis protein tidak terjadi. Terdapat sepuluh jenis asam amino yang terdapat pada Tabel 2.

Tabel 2. Macam-macam Asam Amino (Akram et al., 2011)

No	Essensial Amino Acids	Non-Essensial Amino Acids
1	Lysine	Cysteine
2	Methionine	Tyrosine
3	Valine	Serine
4	Tryptophan	Alanine

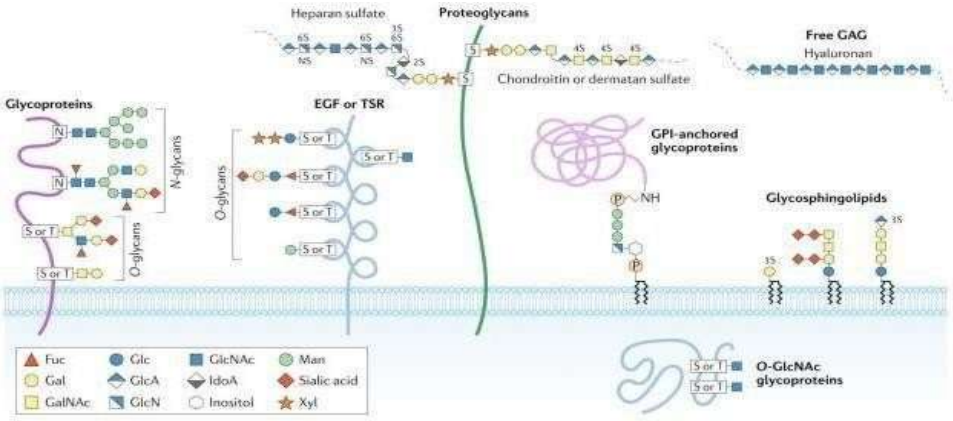
No	Essensial Amino Acids	Non-Essensial Amino Acids
5	Isoleucine	Asparagines
6	Histidine	Aspartic Acid
7	Phenylalanine	Glutamic Acid
8	Threonine	Glycine
9	Leucine	Hydroxylysine
10	Arginine	Proline

2.4. Post Translational Modifications (PTM)

Post Translational Modifications (PTM) sering terjadi di hampir semua protein dan memainkan peran penting dalam berbagai proses biologis dengan secara signifikan mempengaruhi struktur dan dinamika protein. PTM mempunyai peran penting dalam mengatur banyak fungsi biologis sebagai lokalisasi protein dalam sel, stabilitas protein, dan regulasi aktivitas enzimatik (Audagnotto & Dal Peraro, 2017). Macam-macam PTM yaitu *Glycosylation, Phosphorylation, Ubiquitination,* S-nitrosylation, *Lipidation, Acetylation, Proteolysis dan Methylation Alkyl substituents* melekat ke daerah spesifik protein oleh enzim PTM.

2.5. Glikosilasi

Glikosilasi adalah reaksi yang terjadi antara protein dan glukosa pada konsentrasi tinggi, reaksi ini juga sering disebut reaksi *Maillard* (Apriani et al., 2011). Reaksi glikosilasi atau reaksi *Maillard* adalah reaksi antara gugus amina protein dengan gugus aldehid dari glukosa yang dapat membentuk produk-produk reaktif, yang selanjutnya dapat memodifikasi protein. Ciri-ciri reaksi ini dengan terjadinya pencokelatan nonenzimatik antara gula pereduksi dan asam amino bebas yang reaktif dari protein (Suhartono & Setiawan, 2008). Manusia memiliki jenis utama yang sering terjadi, berikut jenis utama glikosilasi manusia pada Gambar 2.



Gambar 2. Jenis Utama Glikosilasi Pada Manusia (Reily et al., 2019).

2.6. Fitur Ekstraksi

Pemrosesan awal data melibatkan transformasi dari pola asli yang dikenal juga sebagai contoh atau objek ke dalam ruang pola yang ditransformasikan. Saat transformasi terjadi terdapat pengurangan dimensi dengan mengekstraksi fitur utama pola (Kittler, 1986). Ekstraksi fitur adalah suatu proses yang penting dalammengklasifikasikan suatu objek untuk menemukan pemetaan fitur yang asli kedalam fitur yang baru. Hal ini dilakukan untuk meningkatkan tingkat akurasi hasil pengklasifikasian suatu objek (Guo et al., 2011). Ekstraksi fitur diproses untuk mengambil dan melihat nilai-nilai yang terdapat pada suatu citra atau data. Proses ekstraksi dilakukan untuk melakukan proses pelatihan (*training*) dan memudahkan pengambilan keputusan**,** seperti cara deteksi pola dan klasifikasi.

Penelitian ini *feature extraction* yang digunakan protein descriptor menggunakan *package* BioSeqClass. Protein Deskriptor adalah tools untuk melakukan proses *feature extraction* pada protein *sequence*, protein deskriptor bisa juga sebagai substruktur lokal dari molekul protein, yang memungkinkan kita untuk membagi masalah asli menjadi satu set subproblem dan, akibatnya, untuk mengusulkan solusi algoritmik yang lebih efisien. Dalam literatur, seseorang dapat menemukan banyak aplikasi konsep deskriptor yang membuktikan kegunaannya untuk wawasan ke struktur protein 3D, tetapi pendekatan yang diusulkan disajikan lebih dari perspektif biologis daripada dari sudut pandang komputasi atau algoritmik. Algoritma yang efisien untuk identifikasi dan

perbandingan struktural deskriptor dapat menjadi komponen penting dari metode untuk penilaian kualitas struktural serta prediksi struktur tersier.

2.6.1 AA*index*

AA*index* adalah database indeks numerik yang mewakili berbagai fisikokimia dan biokimia sifat asam amino dan pasangan asam amino. AA*index* terdiri dari tiga bagian yaitu AA*index*1 untuk indeks asam amino dari 20 nilai numerik, AA*index*2 untuk matriks mutasi asam amino dan AA*index*3 untuk potensi kontak protein statistik. Semua data berasal dari literatur yang diterbitkan (Kawashima et al., 2008) AA*index* memiliki rumus yang dapat dilihat pada Persamaan 1

$$\text{Aa}index(\text{i}) = \frac{\sum_{n=1}^{N} AAIndex_i\,(aa_n)}{N} \qquad (1)$$

Di mana *i* menunjukkan x indeks $AAIndex_{i,}\, aa_n$ asam amino pada posisi *n* dan *N* adalah jumlah residu dalam urutan. AA*Index* yang digunakan pada *feature extraction* ini, yaitu AA*Index*1 yang bertujuan untuk memberikan informasi mengenai urutan *sequence* yang diberikan. AA*Index* memiliki parameter yang berfungsi untuk memanggil properti nama sifat fisikokimia dan biokimia. Jika yang digunakan parameter aa*index*.name=”all”, maka seluruh nama sifatnya akan dipertimbangkan serta setiap baris mewakili fitur dari satu urutan.

2.6.2. *Composition, Transition, dan Distribution* (CTD)

Informasi dari urutan asam amino dapat berfungsi sebagai petunjuk penting untuk memprediksi posisi protein. *Composition, Transition, dan Distribution* (CTD) adalah sebuah fitur yang berfungsi untuk memprediksi lokasi protein yang akan ditargetkan. Fitur ini merupakan atribut asam amino seperti hidrofobisitas, *normalized van der Waals volume*, polaritas, struktur sekunder dan pelarut aksesibilitas urutan protein (Govindan & Nair, 2011).

Composition (C) memiliki rumus yang dapat dilihat pada Persamaan 2.

$$C(i) = \frac{ni}{N} \quad i = 1,2,3 \qquad (2)$$

Di mana *ni* menunujukkan frekuensi grup I dalam barisan, dan N adalah total Panjang barisan.

Transition (T) memiliki rumus yang dapat dilihat pada Persamaan 3.

$$T(ii) = \frac{nij+nji}{N-1} \quad ij = 12,13,23 \qquad (3)$$

Di mana *nij* dan *nji* masing-masing menunjukkan frekuensi transisi dari grup *I* ke *j* dan *j* ke *I* dalam barisan dan *N* adalah Panjang barisan total. *Distribution* berfungsi sebagai penghitung panjang rantai di mana yang pertama, 25, 50, 75 dan 100% dari asam amino yang berada pada tempatnya masing-masing. (Govindan & Nair, 2011).

2.6.3. *Pseudo Amino Acid Composition* (PseAAC)

Pseudo Amino Acid Composition (PseAAC) diusulkan oleh Chou pada tahun 2001 dengan sebutan PseAAC atau Chou's PseAAC (Zare et al., 2015). *Pseudo AminoAcid Composition* (PseAAC) adalah suatu metode yang berfungsi sebagai alat untuk merumuskan urutan biologis dengan vektor, tetapi masih menyimpan informasi urutan-urutanya. PseAAC protein berisi lebih dari 20 faktor yang berbeda. 20 faktor pertama terkait dengan komponen-komponennya seperti komposisi asam amino konservatif, sedangkan faktor tambahan menggabungkan informasi urutan-urutan mereka melalui berbagai metode. Tiga faktor yang sering digunakan untuk menghasilkan berbagai jenis PseAAc seperti komposisi asam amino, faktor bobot dan tingkat korelasi. PseAAc mendukung enam fitur asam amino yaitu hidrofobisitas, hidrofilisitas, massa rantai samping, pK1 (alfa- COOH), pK2 (NH3) dan pI (Titik

isoelektrik). Karakter-karakter ini diterapkan untuk menilai efek dari perbedaan lokasi asam amino di sepanjang urutan peptide (Chou, 2000).

2.7. *Machine Learning*

Machine learning atau pembelajaran mesin adalah pendekatan dalam *Artifical Intelligence* (AI) yang banyak digunakan untuk menggantikan perilaku manusia untuk menyelesaikan masalah. *Machine learning* memiliki dua aplikasi yaitu, klasifikasi dan prediksi. Klasifikasi adalah memilih atau mengklasifikasin suatu obyek berdasarkan ciri tertentu, sedangkan prediksi adalah menerka keluaran dari suatu data masukan berdasarkan data yang sudah dipelajari. Metode *machine learning* yang sering digunakan yaitu system pengambil keputusan, *support vector machine* (SVM) (Ahmad, 2017).

2.7.1. *Machine Learning* Berdasarkan tipe data

Machine learning mempunyai dua tipe yaitu *supervised learning* dan *unsupervised learning* (Ahmad, 2017).

a. *Supervised learning* adalah salah satu tipe algoritma *machine learning* yang menggunakan dataset dikenal sebagai *training dataset* untuk membuat prediksi. *Supervised* dibagi menjadi dua kategori algoritma yaitu *regression* dan *classification*
b. *Unsupervised learning* adalah salah satu tipe algoritma *machine learning* yang digunakan untuk menarik kesimpulan dari *dataset* yang terdiri dari data *labeled response,* metode *unsupervised learning* yang paling sering digunakan adalah analisa *cluster,* pada analisa data untuk mencari pengelompokan data.

2.8. *Random Forest*

Random forest adalah metode berbasis klasifikasi dan regresi dimana terdapat proses agregasi pohon keputusan. Metode ini dipilih karena menghasilkan kesalahan yang lebih rendah, memberikan akurasi yang bagus dalam klasifikasi, dapat menangani data pelatihan yang jumlahnya sangat besar, dan efektif untuk mengatasi data yang tidak lengkap (Primajaya & Sari, 2018).

Random Forest adalah klasifikasi yang terdiri dari beberapa pohon keputusan. Setiap pohon keputusan dibangun dengan menggunakan vektor acak. Pendekatan umum yang digunakan untuk menyisipkan vektor acak dalam pembentukan pohon adalah memilih nilai *F* acak, seperti *F* atribut (fitur) masukan untuk dibagi pada setiap *node* di pohon keputusan yang akan dibentuk (Mambang & Byna, 2017).

Algoritma *random forest* menggunakan sepuluh pohon keputusan yang dibangkitkan secara acak. *Training* data digunakan sebagai data masukan untuk algoritma *random forest* sedangkan *testing* data digunakan untuk menguji atau mengevaluasi *output* atau model yang dihasilkan dari algoritma *random forest* (Primajaya & Sari, 2018).

Algoritma untuk membangun *random forest* memiliki langkah yang dinamakan dengan *bootstrap* (*bag*) yaitu, pada gugus data yang terdiri dari *n* amatan dan *p* peubah penjelas yaitu, melakukan pengambilan contoh acak yang berukuran *n* dengan pemulihan pada gugus data. selanjutnya dengan menggunakan contoh *bootstrap*, pohon dibangun sampai mencapai ukuran terbesar yaitu tanpa pemangkasan.

Implementasi *random forest* memiliki aspek penting yaitu melakukan *bootstrap sampling* untuk membangun pohon prediksi, masing-masing pohon keputusan memprediksi dengan prediktor acak, lalu *random forest* melakukan prediksi dengan mengombinasikan hasil dari setiap pohon keputusan dengan cara *majority vote* untuk klasifikasi (Primajaya & Sari, 2018).

Pembangunan pohon dilakukan dengan menerapkan *random feature ekstraksi* yaitu m peubah penjelas dipilih secara acak dimana m << p, selanjutnya pemilah terbaik dipilih berdasarkan (m) peubah penjelas, kemudian ulangi langkah tersebut sebanyak k kali untuk membuat sebuah *forest* yang terdiri dari k pohon. Tahapan pembuatan model klasifikasi menggunakan algoritma *random forest* dilakukan setelah membuat pemodelan data latih menggunakan package *random forest* di R programming.

Strength (kekuatan) adalah rata-rata atau ekspektasi ukuran kekuatan akurasi pohon tunggal. Nilai *s* yang semakin besar tandanya bahwa akurasi semakin membaik. Nilai *s* memiliki rumus yang dapat dilihat di Persamaan 4 (Breiman, 2001).

$$s = E_{X,Y} mg(X,Y) \tag{4}$$

Rata-rata korelasi ($\bar{p}$) antar pasangan dugaan dari dua pohon tunggal dalam *random forest* dapat dilihat di Persamaan 5 (Breiman, 2001).

$$P = \frac{E\theta.\theta F\,((\theta,)(\theta)(\theta^F))}{E\theta.\theta F(sd(\theta)sd(\theta))} \tag{5}$$

Dimana *p(Θ,Θ')sd(Θ)sd(Θ')* merupakan korelasi antar pohon. Batasan besarnya kesalahan *(ℰRF)* oleh *random forest* dapat dilihat di Persamaan 6.

$$\sum RF \leq \frac{-}{p}\left(\frac{1-s^2}{s^2}\right) \tag{6}$$

Mtry adalah jumlah berbagai prediktor untuk dicoba pada setiap *node* dan *nodesize* adalah ukuran minimum terminal *node*. *Ntree* adalah jumlah *regression tree* (nilai bawaannya adalah 500 pohon) (Azqhandi et al., 2017). Untuk mendapatkan *ntree* yang optimal, pohon harus dibangun sampai errornya kecil dengan cara memiliki korelasi kecil dan memperkuat strength (Fadilah, 2018).

Pemilihan *mtry* dan *ntree* sangat berpengaruh dalam *random forest*. Sebagai contoh, jika *mtry* besar, kemungkinan pada setiap generasi, output pohon regresi mungkin tumbuh terlalu besar. Berikut ini adalah algoritma *random forest* (Hestie et al., 2008).

1. Buat suatu pengambilan sampel *Z* dengan pengembalian (*replacement*) dari suatu ukuran *N* dari gugus data.
2. Pilih *m* variabel secara acak dari *p* variabel, dimana $m \leq p$. Biasanya ukuran *m* terbaik dipilih dari akar kuadrat dari total jumlah *p* variabel, yaitu$|\sqrt{p}|$. Menurut Breiman, nilai *m* juga dapat diperoleh dari 2x nilai akar kuadrat dari total jumlah *p* variabel yang dapat dilihat di Persamaan 7, 8, dan 9.

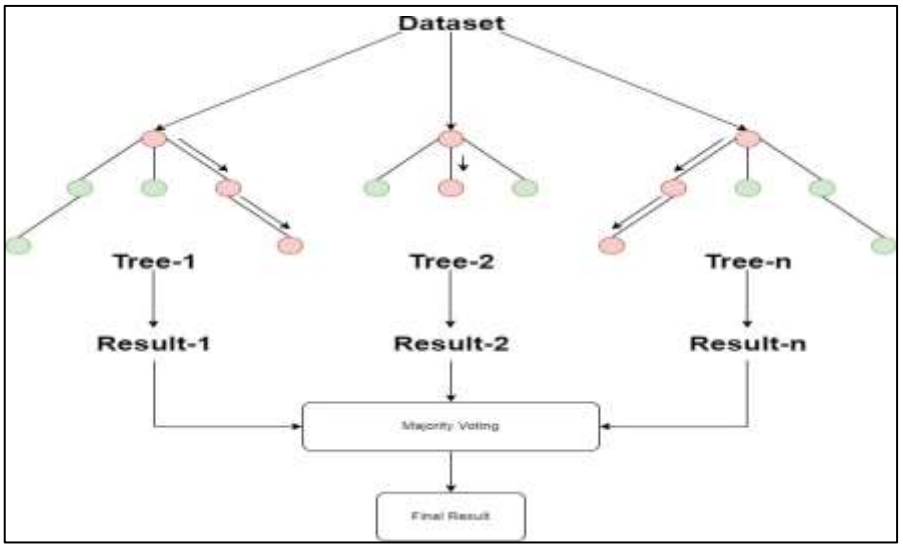
$m = ½ |\sqrt{p}|$ (7)

$m = |\sqrt{p}|$ (8)

$m = 2 \times |\sqrt{p}|$ (9)

Nilai *p* adalah total variabel.

3. Selesai melakukan pemilihan *m* secara acak, selanjutnya pohon ditumbuhkan tanpa pemangkasan (*pruning*).
4. Langkah 1-3 dilakukan sebanyak *n* kali hingga terbentuk suatu *forest* sebanyak *n* pohon.
5. Setelah *forest* terbentuk, selanjutnya mencari nilai misklasifikasi error (*out of bag error*) untuk mendapatkan *mtry* optimal dan diperoleh tingkat kepentingan variabel (*variable importance*) yang lebih stabil. Berikut contoh *random forest* dapat dilihat pada Gambar 3.



Gambar 3. Contoh *random forest*.

Random forest menggunakan gini *index* untuk menentukan kelas akhir disetiap pohon. Pada final class setiap pohon dikumpulkan serta dipilih oleh nilai-nilai weight untuk membangun *classifier* akhir. *Random forest* menggunakan gini *index* diambil dari sistem CART untuk membangun *decision trees*. Berikut adalah contoh perhitungan *random forest*. Dataset yang singkat

dengan menunjukkan pembangunan *single tree* Terdapat 2 atribut, yaitu average_value dan description dapat dilihat pada Tabel 3.

Tabel 3. Contoh soal *Random Forest*.

	Atribut		
Record	***Average Value***	***Description***	***Class***
1	34	3	1
2	22	3	0
3	5	1	1
4	17	4	1
5	10	1	0

Atribut pertama yang akan di split adalah *average_value*. Split yang mungkin untuk *average_value* dalam rentang *node* kiri ialah $5 < x < 34$,dimana x adalah nilai split. Seluruh nilai lain pada setiap dari *node* anak kanan. Berikut split untuk atribut *average_value* dalam dataset adalah:

- *average_value* < 5
- *average_value* < 10
- *average_value* < 17
- *average_value* < 22
- *average_value* < 34

Ambil split pertama untuk melakukan perhitungan yang dapat dilihat pada Tabel 4.

Tabel 4. Perhitungan soal *Random Forest*

	Number Of Record		
	0	**1**	
Average Value <= 5	0	1	N1 = 1
Average Value > 5	2	2	N2 = 4

Kemudian Gini T1, T2, dan gini split dapat dihitung pada Persamaan10, 11, dan 12.

$$Gini(average_value) \leq 5 = 1 - \left(\left(\frac{0}{1}\right)^2 + \left(\frac{1}{1}\right)^0\right) = 0 \quad (10)$$

$$Gini(average_value) \leq 5 = 1 - \left(\left(\frac{2}{4}\right)^2 + \left(\frac{2}{4}\right)^2\right) = 0{,}5 \quad (11)$$

$$Gini\ Split = \left(\frac{1}{5}\right) x\ 0 + \left(\frac{4}{5}\right) x\ 0{,}5 = 0{,}4 \quad (12)$$

Lakukan tahap tersebut untuk mencari gini split <10, <17, <22, dan <34. Sehingga akan mendapatkan keseluruhan gini split yang dapat dilihat pada Tabel 5.

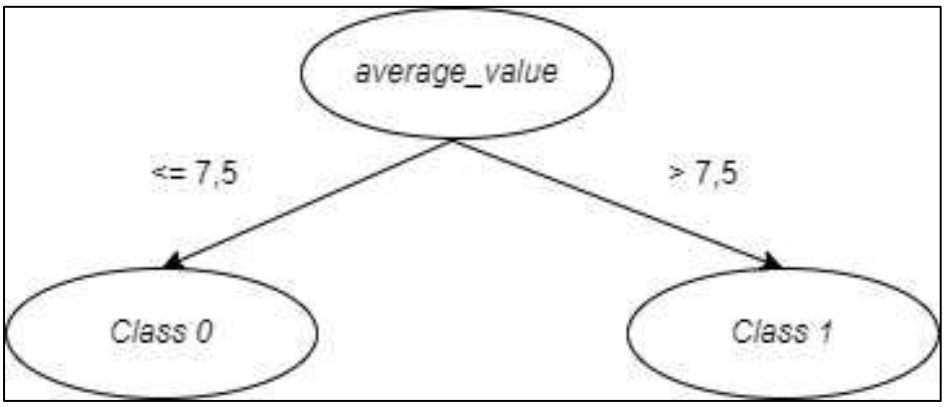
Tabel 5. Nilai gini split pada setiap kemungkinan.

Gini Split	Value
***average_value* <= 5**	**0,4**
average_value <= 10	0,466
average_value <=17	0,466
average_value <= 22	0,41
average_value <=34	0,52

Tabel 5 didapatkan Gini *Index* terendah pada *average_value* <= 5, yaitu 0,4. Dalam *random forest*, split dimana gini *index* terendah dipilih pada nilai split, namun karena nilai-nilai atribut *average_value* bersifat *continue*, titik tengah setiap pasangan nilai berturut dipilih sebagai titik *best* split. *Best* split pada contoh ini dapat dilihat pada Persamaan 13.

$$average_value = \left(\frac{5+10}{2}\right) = 7{,}5 \tag{13}$$

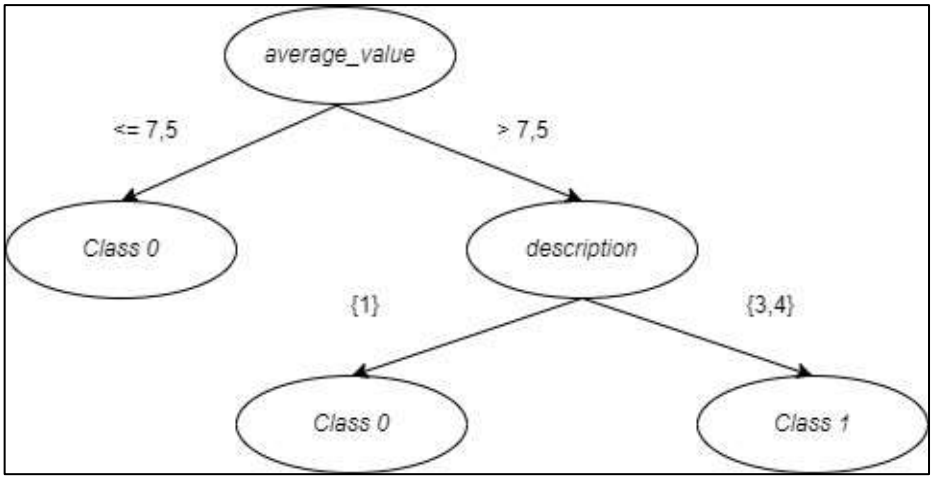
Jadi *best* split bukan pada *average_value* <= 5. Berikut *decision tree* yang pertama dapat dilihat pada Gambar 4.



Gambar 4. *Tree* soal *random forest* variabel pertama.

Langkah-langkah yang telah dilakukan diulangi kembali untuk menentukan sisa atribut pada dataset yang ada untuk mencari Gini *Index* dan

Gini Split. Sehingga *decision trees* akhir akan ditunjukkan seperti *tree* pada Gambar 5



Gambar 5. *Tree* soal *random forest* seluruh variabel.

2.9. Tingkat Misklasifikasi (*Out of Bag Error*)

Salah satu fitur yang ada didalam *random forest* adalah *out-of-bag* (OOB). OOB adalah sepertiga amatan gugus data asli yang tidak termuat dalam contoh pada setiap iterasinya. *Error* OOB bergantung pada korelasi antar pohon dankekuatan (*strength*) masing-masing pohon dalam *random forest* dimana peningkatan korelasi dapat meningkatkan *error* OOB sedangkan peningkatan pohon dapat menurunkan *error* OOB. *Error* OOB dihitung dari proporsi misklasifikasi hasil *random forest* dari seluruh data asli (Breiman, 2001).

2.10. *Variable Importance*

Salah satu tingkat kepentingan variable bebas yang dihasilkan oleh *random forest* adalah *Mean Decrease Accuracy* (MDA), dimana MDA memperlihatkan besar tambahan observasi yang mengalami misklasifikasi jika satu persatu variabel bebas tidak diikut sertakan dalam pengujian. Adapun namanya *Mean Decrease Gini* (MDG), yaitu salah satu ukuran tingkat kepentingan peubah penjelas yang dihasilkan metode *randomforest* (Breiman & Cutler., 2003). Misalkan terdapat p peubah penjelas dengan h = 1,2,…,p maka MDG mengukur tingkat kepentingan peubah penjelas X_h dapat dilihat pada Persamaan 14.

$$MDG_h = \frac{1}{k}\sum[(h,t)(h,t)] \quad (14)$$

(14) Keterangan dari Persamaan 14.

d(h,t) : besar penurunan indeks Gini untuk peubah penjelas. X_h

I(h,t) : memilah simpul *t*.

k : banyaknya pohon dalam *random forest*.

2.10.1. Mean Decrease Accuracy (MDA)

MDA berfungsi untuk seberapa besar akurasi model yang hilang dengan mengecualikan setiap variabel. Semakin mendapatkan akurasi, semakin penting variabel untuk klasifikasi yang sukses. Variabel disajikan dari tingkat kepentingan yang menurun.

2.10.2. *Gini Impruty Index* (GII)

Random forest terdapat sebuah fitur untuk memperlihatkan fitur- fitur yang mempengaruhi hasil akhir dari sebuah klasifikasi. Gini *impruty index* digunakan untuk menentukan *variable importance*. GII bekerja dengan cara memilih fitur secara acak dari kumpulan data yang diklasifikasi secara tidak benar, jika diklasifikasikan secara acak berdasarkan distribusi kelas dari subset. Dalam penelitian ini digunakannya GII untuk me-*rangkin*g 20 fitur teratas yang digunakan dalam klasifikasi *random forest*.

2.11. *Cross Validation*

Cross validation adalah metode evaluasi statistik dan membandingkan algoritma pembelajaran dengan membagi data menjadi dua segmen yaitu digunakan untuk belajar atau melatih model dan yang lainnya digunakan untuk memvalidasi model. Bentuk dasar *cross validation* adalah *k-fold cross-validation*. *K-fold cross-validation*, data pertama kali dipartisi menjadi *k* yang berukuran sama atau lipatan. *Cross validation* yang sering digunakan adalah *10-fold cross-validation* (*k* = 10) *cross validation* digunakan untuk mengevaluasi atau membandingkan algoritma yaitu di setiap iterasi, satu atau lebih banyak algoritma pembelajaran menggunakan *k* 1 lipatan data untuk belajar satu atau

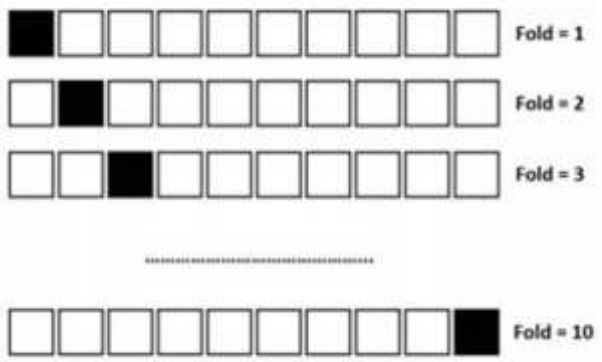
lebih model, dan selanjutnya yang dipelajari model diminta untuk membuat prediksi tentang data di lipatan validasi pada penelitian ini menggunakan *k* 10, *k* 9 digunakan sebagai data trainingdan k 1 di gunakan sebagai data *testing*. Selanjutnya yang dipelajarimodel diminta untuk membuat prediksi tentang data dalam lipatan validasi. *K-fold cross validation* merupakan metode yang membagihimpunan contoh secara acak menjadi *K* himpunan bagian (Ohannessian, 2017).

2.11.1. K-Fold Cross-Validation

K-fold cross-validation data dipartisi pertama ke dalam segmen yang sama (atau hampir sama) atau *fold*. Selanjutnya k iterasi pelatihan dan validasi dilakukan sedemikian rupa sehingga dalam setiap iterasi lipatan yang berbeda dari data diadakan untuk validasi sedangkan sisa *k* 1lipatan digunakan untuk belajar. Data biasanya dikelompokkan sebelum dibagi menjadi *k* lipatan. Stratifikasi adalah proses menata ulang data untuk memastikan setiap lipatan merupakan perwakilan keseluruhan yang baik. Misalnya dalam masalah klasifikasi biner di mana setiap kelas terdiri dari 50% dari data, itu terbaik untuk mengatur data sedemikian rupa sehingga di setiap lipatan, masing-masing kelas terdiri dari sekitar setengah instances. 10-*fold cross-validation* (*k* = 10) adalah yang paling umum. *Cross Validation* digunakan untuk mengevaluasi atau membandingkan belajar algoritma sebagai berikut: di setiap iterasi, satu atau Algoritma pembelajaran lebih banyak menggunakan *k* 1 kali lipat data untuk belajar satu atau lebih model, dan kemudian yang dipelajari model diminta untuk membuat prediksi tentang data dalam lipatan validasi. Kinerja setiap pembelajaran algoritma pada setiap lipatan dapat dilacak menggunakan beberapa yang telah ditentukan metrik kinerja seperti akurasi. Atas selesai, *k* sampel dari metrik kinerja akan tersedia untuk setiap algoritma.

Metodologi yang berbeda seperti rata-rata dapat digunakan untuk mendapatkan agregat mengukur dari sampel ini, atau sampel ini bisa digunakan dalam uji hipotesis statistik untuk menunjukkan itu

satu algoritma lebih unggul dari yang lain (Ohannessian,2017). Ilustrasi *k-fold* untuk *k*=10 dapat dilihat pada Gambar 6.

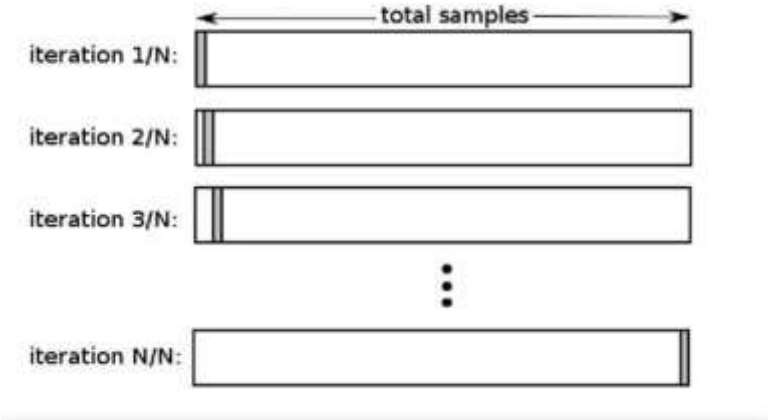


Gambar 6. Ilustrasi *k-fold* untuk *k*=10 (Lukito dan Chrismanto, 2015).

2.11.2. *Leave-One-Out Cross-Validation (LOOCV)*

Leave-one-out cross-validation (LOOCV) adalah spesial kasus *k- fold cross-validation* di mana *k* sama dengan jumlah instance dalam data. Dengan kata lain di setiap iterasi hampir semua data kecuali untuk observasi tunggal digunakan untuk pelatihan dan modelnya diuji pada pengamatan tunggal itu. Estimasi akurasi diperoleh menggunakan LOOCV diketahui hampir tidak ada bias tetapi memiliki varians yang tinggi, menyebabkan tidak dapat diandalkan perkiraannya. Model ini masih banyak digunakan ketika tersedia data sangat jarang, terutama di bioinformatika di mana hanya lusinan sampel data yang tersedia (Refaeilzadeh, Tang dan Liu, 2008). LOOCV adalah bentuk khusus dari *cross- validation*, yaitu jumlah *fold* sama dengan jumlah data training. LOOCV ini sering digunakan untuk memperkirakan kemampuan generalisasi penggolongan statistik (yaitu kinerja pada data yang sebelumnya tidak terlihat) (Cawley dan Talbot, 2003). Pendekatan ini meninggalkan 1 titik data dari data pelatihan, yaitu jika ada n titik data dalam sampel asli maka, sampel *n*-1 digunakan untuk melatih model dan poin *p* digunakan sebagai set validasi. Ini diulang untuk semua kombinasi di mana sampel asli dapat dipisahkan dengan cara ini, dan kemudian kesalahan dirata- ratakan untuk semua percobaan, untuk memberikan

keefektifan secara keseluruhan. Ilustrasi *Leave-One-Out Cross-Validation* dapat di lihat pada Gambar 7.



Gambar 7. Ilustrasi *Leave-One-Out Cross-Validation* (Cawley dan Talbot 2003).

2.11.3. *Hold-Out Validation*

Hold-out cross-validation data dibagi menjadi dua dataset yang berbeda yang diberi label sebagai pelatihan dan dataset pengujian Keuntungan dari metode ini adalah itu proporsi dari ketiga himpunan data ini tidak dibatasi secara ketat. Dalam hal ini, ada kemungkinandistribusi yang tidak merata dari berbagai kelas data ditemukan dalam pelatihan dan dataset uji. Untuk memperbaikinya, dataset pelatihan dan uji dibuat dengan distribusi kelas data yang berbeda. Proses ini disebut stratifikasi (Reitermanov, 2010). Untuk menghindari *over-fitting* set tes *independen* lebih disukai.

Pendekatan alami adalah membagi data yang tersedia menjadi dua bagianyang tidak tumpang tindih: satu untuk pelatihan dan yang lain untuk pengujian. Data tes diadakan dan tidak melihat selama pelatihan.

Masa berlaku validasi penonaktifan tumpang tindih antara data pelatihan dan data uji, menghasilkan perkiraan yang lebih akurat untuk generalisasi kinerja algoritma. Kelemahannya adalah itu prosedur ini tidak menggunakan semua data yang tersedia dan hasilnya sangat tergantung pada pilihan untuk pelatihan / tes terpisah. Instansi yang dipilih untuk dimasukkan dalam set tes mungkin terlalu mudah atau terlalu sulit untuk digolongkan dan ini dapat mengubah hasilnya. Selanjutnya, data dalam set tes mungkin berharga untuk pelatihan dan jika itu diadakan kinerja prediksi, kembali memimpin untuk hasil yang miring. Masalah- masalah ini bisa sebagian ditangani dengan mengulang beberapa validasi hold- out kali dan rata-rata hasil, tetapi kecuali pengulangan ini dilakukan secara sistematis, beberapa data dapat dimasukkan dalam set tes beberapa kali sementara yang lain tidak termasuk sama sekali, atau sebaliknya beberapa data mungkin selalu jatuh dalam set tes dan tidak pernah mendapat kesempatan berkontribusi pada fase pembelajaran. Untuk menghadapi ini menantang dan memanfaatkan data yang tersedia secara maksimal (Ohannessian, 2017).

2.12. *Confusion matrix*

Confusion matrix adalah metode yang digunakan untuk menghitung performa model klasifikasi yang berisi aktual dan prediksi. Menurut Kuhn & Johnson (2013) *confusion matrix* memiliki perhitungan matriks yang digunakan untuk mengukur performa dari prediktor. Perhitungan *confusion matrix* dapat dilihat pada Tabel 6.

Tabel 6. *Confusion Matrix* (Kuhn & Johnson, 2013).

Predict	***Actual***	
	Positive	***Negative***
Positive	***TP***	***FP***
Negative	***FN***	***TN***

Adapun istilah yang digunakan pada *confusion matrix* yang berada pada Tabel 6, diantaranya:

a. *True Positive* (TP): jumlah data positif yang diklasifikasikan dengan benar oleh sistem.
b. *True Negative* (TN): jumlah data negatif yang diklasifikasikan dengan benar oleh sistem.
c. *False Positive* (FP): jumlah data positif yang diklasifikasikan dengan salah oleh sistem.
d. *False Negative* (FN): jumlah data negatif yang diklasifikasikan dengan salah oleh sistem.

Confusion matrix memiliki hasil statistik untuk penilaian kinerja. Hasil tersebut diantaranya akurasi (ACC), sensitivitas (SE), spesifisitas (SP), dan *MatthewCorrelation Coefficient* (MCC).

a. *Accuracy* (ACC)

Akurasi adalah matrik yang paling umum untuk evaluasi pengklasifikasian. Akurasi memperkirakan probabilitas nilai sebenarnya dari label kelas (Bekkaret al., 2013). Perhitungan akurasi dapat dilihat pada Persamaan 15.

$$ACC = \frac{TP + TN}{TP + FP + FN + TN} \tag{15}$$

b. *Sensitivity* (SN)

Sensitivitas adalah kelengkapan atau keakuratan informasi positif yang didapat oleh sistem dengan jumlah keseluruhan informasi yang ada (Bekkar et al., 2013). Perhitungan sensitivitas dapat dilihat pada Persamaan 16.

$$SN = \frac{TP}{TP + FP} \tag{16}$$

c. *Specificity* (SP)

Spesifitas adalah probabilitas dari informasi proporsi negatif yang diberikan oleh kelas sekunder (Bekkar et al., 2013). Perhitungan spesifisitas dapat dilihat pada Persamaan 17.

$$SP = \frac{TN}{TN + FP} \tag{17}$$

d. *Matthew Correlation Coefficient* (MCC)

Matthew Correlation Coefficient (MCC) adalah perhitungan klasifikasi yangbernilai positif dan negatif yang bernilai benar dan salah (Bekkar et al., 2013). Perhitungan MCC dapat dilihat pada Persamaan 18.

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FN)}(TP + FP)(TN + FP)(TN + FN)} \quad (18)$$

III. METODOLOGI PENELITIAN

3.1. Tempat dan Waktu Penelitian

3.1.1 Tempat Penelitian

Penelitian ini dilakukan di Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung yang beralamatkan di Jalan Soemantri Brojonegoro No. 1 Gedung Meneng, Bandar Lampung.

3.1.2. Waktu Penelitian

Penelitian ini dilaksanakan mulai dari semester genap tahun ajaran 2020/2021 sampai bulan November 2022.

Berikut adalah ringkasan dari alokasi penelitian ini sebagai berikut:

a. Pengambilan atau Pengumpulan Data

   Data *benchmark* telah dibuat oleh Yun Xu et al. (2016) dan telah dipublikasikan. Data penelitian ini dapat diakses melalui link http://app.aporc.org/Gly- PseAAC/.

b. *Preprocessing Data*

   Dilakukan bertujuan untuk mengurangi dan menghilangkan data yang berulang atau data yang sama.

c. *Feature extraction*

   Dilakukan *feature extraction* menggunakan CTD, PseAAC, danAa*index*.

d. *k-fold Cross Validation*

   Menggunakan 10-*fold* dengan membagi dataset menjadi 10% data *testing*, 90% data *training*.

e. Pembentukan Model Data

   Penelitian ini dibuat dengan sebuah model *random forest*

3.2. Data dan Alat

3.2.1. Data

Data yang di gunakan pada penelitian ini yaitu data *sequence* protein *lysine* yang didapatkan dari *study literature* penelitian Yun Xu et al. (2016) yang didapatkan dari sumber: http://app.aporc.org/Gly- PseAAC/. Setiap sequence memiliki panjang 15 *sequence*. Percobaan *dataset* padapenelitian Yun Xu et al. dapat dilihat pada Tabel 7.

Tabel 7. Data *Sequence* Protein Glikosilasi.

Data	*Dataset*	Jumlah *Sequence* Protein
Benchmark	*Positif*	214
	Negative	406

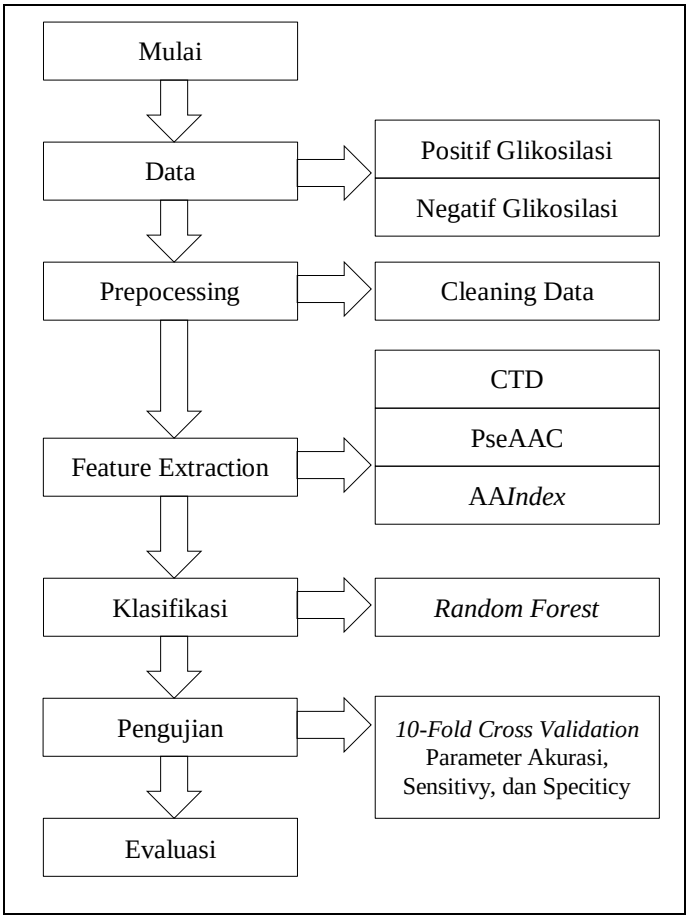
3.2.2. Alat

Dalam penelitian ini peneliti menggunakan *Software* dan *Hardware* sebagai berikut :

a. Penelitian menggunakan *software* sistem operasi sebagai berikut untuk mendukung kinerja seperti Windows 10 64-bit sebagai sistem operasi, Microsoft Excel 2010, R studio versi 3.6.1 dan R versi 3.6.1.
b. *Hardware* : Prosesor Intel® Celeron CPU N3050 @ 1.60 GHz, VGA NVIDIA® NVS 3100M, DDR3L, RAM 4 GB, Memory HDD 250 GB.

3.3. Alur Kerja Penelitian

Penelitian ini melalui beberapa tahapan yang dapat dilihat pada Gambar 8 dilakukan dalam penelitian. Pada Gambar 8 menampilkan bagaimana cara untuk melakukan prediksi dengan menggunakan metode *random forest*.



Gambar 8. Tahapan proses klasifikasi dengan model *Random Forest*.

Berdasarkan diagram alur metodologi yang ditampilkan pada Gambar 8, dibawah inimerupakan penjelasan dari setiap proses prediksi dengan model *random forest*:

3.3.1. Data

Data yang diperlukan yaitu *sequence* protein lisin yang akan digunakan untuk melakukan Klasifikasi Glikosilasi Pada *Sequence* Protein Lisin Menggunakan *Random Forest*. Data yang akan digunakan dalam penelitian ini didapatkan dari *study literature* penelitian Yun Xu et al. (2016). Data penelitian terdiri dari kelas negative dan kelas positif.

3.3.2. *Preprocessing*

Preprocessing bertujuan untuk mengurangi dan menghilangkan data yang berulang atau data yang sama. Tahapan *cleaning* pada *sequence* protein agar data yang akan diklasifikasikan tidak terjadi banyak kemiripan.

3.3.3. *Feature Extraction*/ Ekstraksi fitur

Ekstraksi fitur merupakan proses pengambilan ciri yang terdapat didalam objek yang ada dalam gambar. Proses ekstraksi fitur bertujuan untuk mengambil atau mengekstraksi nilai-nilai unik dari suatu objek yang membedakan dengan objek yang lain. Pada penelitian ini menggunakan metode protein deskriptor adalah substruktur lokal dari molekul protein, yang memungkinkan untuk membagi masalah menjadi satu set subproblem dan akibatnya untuk mengusulkan solusi algoritmik yang lebih efisien. *Package* BioSeqClass dengan menggunakan 3 macam protein diskriptor yaitu CTD, PseAAC, dan AA *index*.

3.3.4. *Classification*

Metode klasifikasi pada penelitian ini menggunakan *random forest* terdapat 2 parameter yang digunakan pada klasifikasi random forest yaitu, *mtry* dan *ntree*. Nilai *mtry* yang digunakan ialah 4, 8, 16 dan *ntree* yang digunakan adalah 250, 500, 750, dan 1000.

3.3.5. Pengujian

K-fold cross validation digunakan untuk melakukan pengujian, pada pengujian *cross validation* banyak menggunakan *k* 1 kali lipat data untuk belajar satu atau lebih model, pada penelitian ini menggunakan *k* 10, *k* 9 digunakan sebagai data *training* dan *k* 1 di gunakan sebagai data *testing*.

3.3.6. Evaluasi

Evaluasi ini berasal dari *k-fold cross-validation* dan *random forest* yang menghasilkan nilai *confussion matrix* dengan nilai akurasi, sensitifitas, spesifisitas dan *Matthew Correlation Coefficient*.

V. PENUTUP

5.1. Simpulan

Berdasarkan hasil analisis dan pembahasan pada bab empat mengenai klasifikasi glikosilasi algoritme *Random Forest* dapat disimpulkan sebagai berikut:

1. Data protein glikosilasi yang masih bertipe data *string* diubah terlebih dahulu menggunakan tiga fitur ekstraksi yang terdiri dari AA*index, composition, transition, dan distribution* (CTD), dan *pseudo amino acid composition* (PseAAC).
2. Glikosilasi di klasifikasikan dengan menggunakan *random forest* menggunakan parameter *mtry* 4, 8, dan 16 serta *ntree* 250, 500, 750 dan 1000.
3. Didapatkan hasil tertinggi menggunakan *ntree* 4 dan *mtry* 500 dengan nilai akurasi 67,86%, sensitivitas 4,21%, spesifisitas 97,20%, dan MCC 2,87%.
4. Setelah dilakukannya perbandingan dengan penelitian sebelumnya, ternyata penelitian (Yun Xu et al., 2016) yang menggunakan metode SVM memiliki nilai akhir yang baik. Dengan ini membuktikan bahwa menggunakan *random forest* masih memiliki hasil akhir yang kecil disaat sudah dipengaruhi oleh *feature extraction* serta parameter yang digunakan.

5.2. Saran

Berdasarkan penelitian yang dilakukan maka diperoleh beberapa saran untuk pengembangan penelitian ini lebih lanjut sebagai berikut:

1. Pada penelitian selanjutnya diharapkan agar menggunakan metode klasifikasi lainnya, misalnya K-Nearest Neighbors (KNN) untuk melihat hasil klasifikasi yang berbeda, untuk melihat perbandingan hasil dan kualitas kinerja klasifikasi.
2. Penelitian ini dapat menggunakan *feature extraction* lain seperti *hydrophobicity*, *quasi-sequence-order* (QSO), dan lain-lain.

DAFTAR PUSTAKA

Ahmad, A. 2017. Mengenal Artificial Intelligence, Machine Learning, Neural Network, dan Deep Learning. *Jurnal Teknologi Indonesia*, (October), 3.

Apriani, N., Suhartono, E., Akbar, I. Z., & Mangkurat, U. L. 2011. Korelasi Kadar Glukosa Darah dengan Kadar Advanced Oxidation Protein Products (AOPP) Tulang pada Tikus Putih Model Hiperglikemia. *Jkm*, *11*(Juli), 48–55.

Audagnotto, M., & Dal Peraro, M. 2017. Protein post-translational modifications:In silico prediction tools and molecular modeling. *Computational and Structural Biotechnology Journal*, *15*, 307–319.https://doi.org/10.1016/j.csbj.2017.03.004.

Bekkar, M., Djemaa, H. K., & Alitouche, T. A. 2013. Evaluation Measures for Models Assessment over Imbalanced Data Sets. *Journal of Information Engineering and Applications*, *3*(10), 27–38. Retrieved from http://www.iiste.org/Journals/*index*.php/JIEA/article/view/7633.

Katili, A. S. 2009. Struktur Dan Fungsi Protein Kolagen. *Jurnal Pelangi Ilmu*, *2*(5), 19–29.

Mambang, & Byna, A. 2017. Analisis Perbandingan Algoritma C.45, Random Forest Dengan Chaid Decision Tree Untuk Klasifikasi Tingkat Kecemasan Ibu Hamil. *Seminar Nasional Teknologi Informasi Dan Multimedia*, 103–108.

Mipa, F., Semarang, U. N., Mulya, N. S., Lemong, K., & Barat, P. 2017. ProfilProtein Susu Dan Produk Olahannya. *Jurnal MIPA*, *39*(2), 98–106.

Ohannessian, S. 2017. Historical background. *Language in Zambia*, 271–291.https://doi.org/10.4324/9781315106786-12

Pascovici, D., Wu, J. X., McKay, M. J., Joseph, C., Noor, Z., Kamath, K., Mirzaei, M. 2019. Clinically relevant post-translational modification analyses—maturing workflows and bioinformatics tools. *International*

Journal of Molecular Sciences, *20*(1). https://doi.org/10.3390/ijms20010016

Primajaya, A., & Sari, B. N. 2018. Random Forest Algorithm for Prediction of Precipitation. *Indonesian Journal of Artificial Intelligence and Data Mining*, *1*(1), 27. https://doi.org/10.24014/ijaidm.v1i1.4903

Rosandy, T. 2016. PERBANDINGAN METODE NAIVE BAYES CLASSIFIER DENGAN METODE DECISION TREE (C4.5) UNTUK MENGANALISA KELANCARAN PEMBIAYAAN (Study Kasus : KSPPS / BMT AL-FADHILA. *Jurnal Teknologi Informasi Magister Darmajaya*, *2*(01), 52–62

Suhartono, E., & Setiawan, B. 2008. Modifikasi Protein Akibat Pembebanan Glukosa dengan Model Reaksi Glikosilasi Nonenzimatik in vitro Protein Modification of Glucose Effect of Encumbering with the Model React. *Jurnal Ilmiah*, *08*, 40–47.

Akram, M., Asif, H. M., Uzair, M., Akhtar, N., Madni, A., Ali Shah, S. M., Hasan, Z. U., & Ullah, A. (2011). Amino acids: A review article. *Journal of MedicinalPlants Research*, *5*(17), 3997–4000.

Guo, L., Rivero, D., Dorado, J., Munteanu, C. R., & Pazos, A. (2011). Automatic feature extraction using genetic programming: An application to epileptic EEG classification. *Expert Systems with Applications*, *38*(8), 10425–10436. https://doi.org/10.1016/j.eswa.2011.02.118

Kawashima, S., Pokarowski, P., Pokarowska, M., Kolinski, A., Katayama, T., & Kanehisa, M. (2008). AA*index*: Amino acid *index* database, progress report 2008. *Nucleic Acids Research*, *36*(SUPPL. 1), 202–205. https://doi.org/10.1093/nar/gkm998